

WHAT IS CLAIMED IS:

1. A method for processing one or more data objects in business applications, comprising:

creating an electronic data element comprising a first data field and a second data field, wherein the first data field contains data representing an identifier functioning as a link to one or more data objects and the second data field represents a state of the identifier in the first data field; and

associating the identifier of the electronic data element with the one or more data objects.

2. The method of claim 1, wherein the second data field may be one of:

a) a first state, in which said electronic data element may be accessed by one or more data object processing operations and whereby said identifier is assignable to one or more data objects,

b) a second state, in which said electronic data element may not be accessed by one or more data object processing operations and whereby said identifier is assignable to one or more data objects, or

c) a third state, in which said electronic data element may not be accessed by one or more data object processing operations and whereby said identifier is not assignable to one or more data objects.

3. The method of claim 1, wherein the first data field and the second data field are in a table.

4. The method of claim 1, wherein the first data field is in a first table and the second data field is in a second table.

5. The method of claim 1, wherein the electronic data element is implemented in object orientated programming as an instance of a class.

6. The method of claim 1, wherein the data structure further comprises a third data field functioning as a flag representing whether the electronic data element is the default identifier.

7. The method of claim 2, further comprising changing the first data field from the first state to the second state.

8. The method of claim 7, further comprising storing the one or more data objects.

9. The method of claim 8, further comprising changing the second field to the third state if the one or more assigned data objects are committed.

10. The method of claim 9, further comprising creating a new electronic data element and setting the second field of the new electronic data element to the first state.

11. The method of claim 10, wherein the data structure further comprises a third data field functioning as a flag representing whether the electronic data element is the default identifier, and further comprising flagging third data field of the new electronic data element as the default identifier.

12. The method of claim 10, further comprising setting the second field of the prior electronic data element to the second state.

13. The method of claim 10, further comprising examining the third field of a previous electronic data element, and, if the third field of the previous electronic

data element is flagged as the default identifier, the third field of the previous electronic data element is flagged as not being the default identifier.

14. The method of claim 2, further comprising:
setting a block on the electronic data element;
examining the second field of the electronic data element; and
if the state of the second field of the electronic data element is the first state or the second state, preventing a change in the state of the second field to the third state.
15. The method of claim 14, removing the block if storing of the data object is committed.
16. The method of claim 14, further comprising irreversibly setting the block if the electronic data element is in the third state.
17. The method claim 1, further comprising share locking the electronic data element.
18. The method of claim 17, further comprising share locking the electronic data element prior to association of the electronic data element to the one or more data objects.
19. The method of claim 17, further comprising unlocking the share locking of the electronic data element after storing of the data object is committed.
20. The method of claim 17, further comprising examining the state of the share lock of the data element prior to association of the electronic data element to the one or more data objects.

21. The method of claim 1, further comprising replicating the data objects from a source system to a target system.

22. The method of claim 1, wherein the identifier of the first data field comprises a globally unique identifier.

23. The method of claim 1, wherein the identifier of the first data field comprises a time stamp.

24. A system for processing one or more data objects in business applications, comprising:

a memory; and

a microprocessor coupled to the memory and programmed to:

create an electronic data element comprising a first data field and a second data field, wherein the first data field contains data representing an identifier functioning as a link to one or more data objects and the second data field represents a state of the identifier in the first data field; and

associate the identifier of the electronic data element with the one or more data objects.

25. The system of claim 24, wherein the second data field may be one of:

a) a first state, in which said electronic data element may be accessed by one or more data object processing operations and whereby said identifier is assignable to one or more data objects,

b) a second state, in which said electronic data element may not be accessed by one or more data object processing operations and whereby said identifier is assignable to one or more data objects, or

c) a third state, in which said electronic data element may not be accessed by one or more data object processing operations and whereby said identifier is not assignable to one or more data objects.

26. The system of claim 24, wherein the first data field and the second data field are in a table.

27. The system of claim 24, wherein the first data field is in a first table and the second data field is in a second table.

28. The system of claim 24, wherein the electronic data element is implemented in object orientated programming as an instance of a class.

29. The system of claim 24, wherein the data structure further comprises a third data field functioning as a flag representing whether the electronic data element is the default identifier.

30. The system of claim 25, wherein the microprocessor is further programmed to change the first data field from the first state to the second state.

31. The system of claim 30, wherein the microprocessor is further programmed to store the one or more data objects.

32. The system of claim 31, wherein the microprocessor is further programmed to change the second field to the third state if the one or more assigned data objects are committed.

33. The system of claim 32, wherein the microprocessor is further programmed to create a new electronic data element and set the second field of the new electronic data element to the first state.

34. The system of claim 33, wherein the data structure further comprises a third data field functioning as a flag representing whether the electronic data element is the default identifier, and wherein the microprocessor is further programmed to flag third data field of the new electronic data element as the default identifier.

35. The system of claim 33, wherein the microprocessor is further programmed to set the second field of the prior electronic data element to the second state.

36. The system of claim 33, wherein the microprocessor is further programmed to examine the third field of a previous electronic data element, and, if the third field of the previous electronic data element is flagged as the default identifier, the third field of the previous electronic data element is flagged as not being the default identifier.

37. The system of claim 25, wherein the microprocessor is further programmed to:

set a block on the electronic data element;

examine the second field of the electronic data element; and

if the state of the second field of the electronic data element is the first state or the second state, prevent a change in the state of the second field to the third state.

38. The system of claim 37, wherein the microprocessor is further programmed to remove the block if storing of the data object is committed.

FINNEGAN
HENDERSON
FARABOW
GARRETT &
DUNNER LLP

1300 I Street, NW
Washington, DC 20005
202.408.4000
Fax 202.408.4400
www.finnegan.com

39. The system of claim 37, wherein the microprocessor is further programmed to irreversibly set the block if the electronic data element is in the third state.

40. The system claim 24, wherein the microprocessor is further programmed to share lock the electronic data element.

41. The system of claim 40, wherein the microprocessor is further programmed to share lock the electronic data element prior to association of the electronic data element to the one or more data objects.

42. The system of claim 40, wherein the microprocessor is further programmed to unlock the share locking of the electronic data element after storing of the data object is committed.

43. The system of claim 40, wherein the microprocessor is further programmed to examine the state of the share lock of the data element prior to association of the electronic data element to the one or more data objects.

44. The system of claim 24, wherein the microprocessor is further programmed to replicate the data objects from a source system to a target system.

45. The system of claim 24, wherein the identifier of the first data field comprises a globally unique identifier.

46. The system of claim 24, wherein the identifier of the first data field comprises a time stamp.

FINNEGAN
ENDERSON
FARABOW
GARRETT &
DUNNER LLP

1300 I Street, NW
Washington, DC 20005
202.408.4000
Fax 202.408.4400
www.finnegan.com